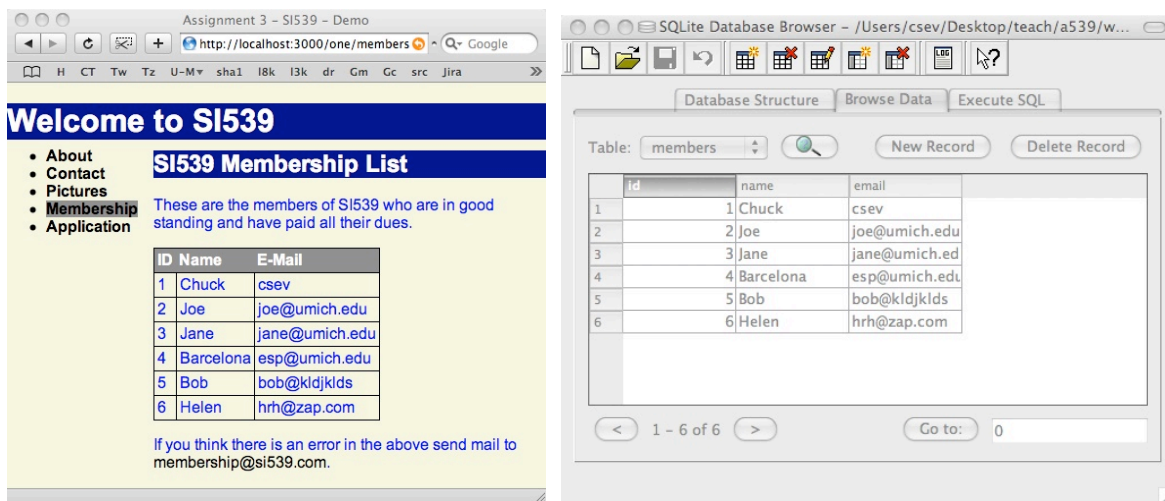


Un-Assignment 7A – Reading from the Database

Un-Due Date: Tuesday March 18, 2008 at 11:55PM

**** This Assignment is Optional **** If you are going to do this assignment you should complete it before the Due date above so you can focus on the next assignment. There will be no place in CTools to hand anything in and nothing will be graded. However you are welcome to ask for questions and to get help on this assignment like any other assignment.

In this Assignment you will replace the hard-coded data in the table on the membership page with the objects pulled from the database table.



The left screenshot shows a web browser window titled 'Assignment 3 - SI539 - Demo' with the URL 'http://localhost:3000/one/members'. It displays a 'Welcome to SI539' page with a sidebar menu (About, Contact, Pictures, Membership, Application) and a 'SI539 Membership List' section. The list contains a table with 6 members: Chuck, Joe, Jane, Barcelona, Bob, and Helen. The right screenshot shows a 'SQLite Database Browser' window with the 'members' table selected. The table has columns 'id', 'name', and 'email' and contains the same 6 rows of data as the web page.

ID	Name	E-Mail
1	Chuck	csev
2	Joe	joe@umich.edu
3	Jane	jane@umich.edu
4	Barcelona	esp@umich.edu
5	Bob	bob@kldjklds
6	Helen	hrh@zap.com

You do not have to make a copy of Assignment 7 - simply modify it in place. Add the following to your members action in your controller:

```
class OneController < ApplicationController
  def index
    end

  def members
    @members = Member.find(:all)
    logger.info "Members method"
    logger.info @members
  end
end
```

Keep everything else you have in your controller - just add the new code to the members method. This code retrieves "all" of the member objects from the members table and puts the list / array of those objects into the instance variable @members.

In your members.rhtml view file replace your hard-coded table with code like the following:


```

<table>
  <tr>
    <th>ID</th>
    <th>Name</th>
    <th>E-Mail</th>
  </tr>
  <% @members.each do |valencia| %>
    <tr>
      <td><%= valencia.id %></td>
      <td><%= valencia.name %></td>
      <td><%= valencia.email %></td>
    </tr>
  <% end %>
</table>

```

The code in bold loops through the list of members in the @members instance variable. The iteration variable for the loop is "Valencia" - this will be each successive member - for each time the loop is executed. Note carefully when the <% and <%= are used. Sometimes we just run Ruby code and other times we run Ruby code and produce output.

If you view source on the generated page it will look as follows:

```

<table>
  <tr>
    <th>ID</th>
    <th>Name</th>
    <th>E-Mail</th>
  </tr>
  <tr>
    <td>1</td>
    <td>Chuck</td>
    <td>csev</td>
  </tr>
  <tr>
    <td>2</td>
    <td>Joe</td>
    <td>joe@umich.edu</td>
  </tr>
  <tr>
    <td>3</td>
    <td>Jane</td>
    <td>jane@umich.edu</td>
  </tr>
  [ .. snip ...]
</table>

```

The HTML is simply a set of table rows - with a row generated each time through the loop. Viola! You are done - no screen shots - no due date - just fun, cool reading from the database and writing a loop in Ruby.